In the world of business, leaders must maximize efficiency, eliminate waste, even out roadblocks, and keep productions flowing in accordance with demand.

Once a business venture takes flight, however, a lot of different people get involved, each with their own knowledge, talent and methods for getting things accomplished. In order to ensure flow and consistency from party to party, methods must be in place that keep productions moving along.

The following article discusses and compares some of the most widespread methods and frameworks used in the world of development and manufacturing here in the 21st century. Read along to explore different Agile methodologies and discover which one is right for your project.

**I. Agile**



The Agile methodology derives from a namesake [manifesto](http://agilemanifesto.org/), which advanced ideas that were developed to counter the more convoluted methods that pervaded the software development world despite being notoriously inefficient and counterproductive. Promoting similar values to Lean, Agile methods mostly focus on the following key principles:

* Satisfying customers is of foremost importance
* Develop projects with inspired contributors
* Interactions are best when done in person
* Software that works is a measure of progress
* Reflect and adapt on an ongoing basis

Additionally, the four core values of Agile are as follows:

* Individuals and interactions over processes and tools
* Working software over comprehensive documentation
* Customer collaboration over contract negotiation
* Responding to change over following a plan

Regardless of which Agile methodology a team adopts, the benefits cannot be fully realized without the commitment of everyone involved. So, here are some of the most popular Agile methodology examples.

**II. Extreme Programming (XP)**

One of the foremost Agile methodologies is called Extreme Programming (XP), which involves a high degree of participation between two parties in the software exchange: customers and developers. The former inspires further development by emphasizing the most useful features of a given software product through testimonials. The developers, in turn, base each successive set of software upgrades on this feedback while continuing to test new innovations every few weeks.

XP has its share of pros and cons. On the upside, this Agile methodology involves a high level of collaboration and a minimum of up-front documentation. It’s an efficient and persistent delivery model. However, the methodology also requires a great level of discipline, as well as plenty of involvement from people beyond the world of information technology. Furthermore, in order for the best results, advanced XP proficiency is vital on the part of every team member.



Essentially, XP works best among small teams comprised of experienced developers with a proven track record in communication and management.

**III. SCRUM**

SCRUM is an Agile methodology that consists of a more complex set of development principles. They focus on the management aspects of projects. Under this method, the owner of a given product will collaborate with information technology teams and businesses to collectively isolate and harness the functions of a system in what is known as a product backlog. Assorted team members will then distribute a software program in increments — this is called a **sprint**, and the typical duration is one month. Upon delivery, an analysis is made of the **product backlog**, and **priorities** are shifted if necessary with each repetition of the cycle.



As with XP, implementing SCRUM offers a mix of benefits and disadvantages. On one hand, this Agile methodology enables management teams to spot problems at the development stage — SCRUM also promotes transparency among colleagues. However, SCRUM can also result in slapdash programming and leave scant records for handover. As such, the method is most appropriate for information technology businesses that focus on products, features and delivery in partnership with management teams. A good practice from SCRUM is the [daily stand-up meetings](https://kanbanize.com/blog/running-a-better-stand-up-meeting/) – a timeboxed daily event, where all team members discuss work progress and possible obstacles.

SCRUM is one of the frameworks that revolutionized the software development industry. It became popular because of its fast iterations and active collaboration between teams, customers, and stakeholders. For the sake of better collaboration, there are predefined team roles:

* **Product Owner.** The PO is responsible for understanding the business and market requirements. After this, she/he need to prioritize work, build a product backlog and make sure that everyone understands the work items.
* **Scrum Master.** The SM educates the team, the product owner, and the business on scrum processes. It’s her/his responsibility to manage the workflow of the team and to schedule all resources needed for the completion of each task.
* **Scrum Team.**  The team usually consists of people with different skills such as developers, automation engineers, and testers. All team members have to support each other in order to be successful. Most efficient scrum teams are usually co-located and with the size of 5 to 8 members.

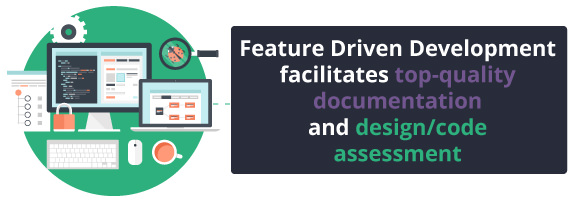
People who’ve compared XP and SCRUM have said that both operate in brief iterations for the advancement of collective work, which involves businesses managers, development teams and test subjects throughout the span of a project.

**IV. Feature-Driven Development (FDD)**

Centered on the developer, the FDD methodology involves turning models into builds at fortnightly iterations. In contrast to XP and SCRUM, FDD centers on strict operations that involve the walkthrough of domains, as well as design, code and inspection.

The model is then built up along with a list of features. For every feature, a development and design plan is implemented. Following a series of inspections, a unit test is performed to see whether it’s ready for the build stage.

FDD advocates a strict organization of the building process, which results in viable software that can be produced on a consistent basis. On the plus side, FDD facilitates top-quality documentation and design/code assessment. However, the method demands an advanced level of design skill and planning foresight — early imperfections can lead to prolonged corrections. FDD is most optimal for big business developers in the banking and financial sectors, where process maturity and [quality control](https://kanbanize.com/blog/difference-between-quality-control-and-continuous-improvement) are obligatory.



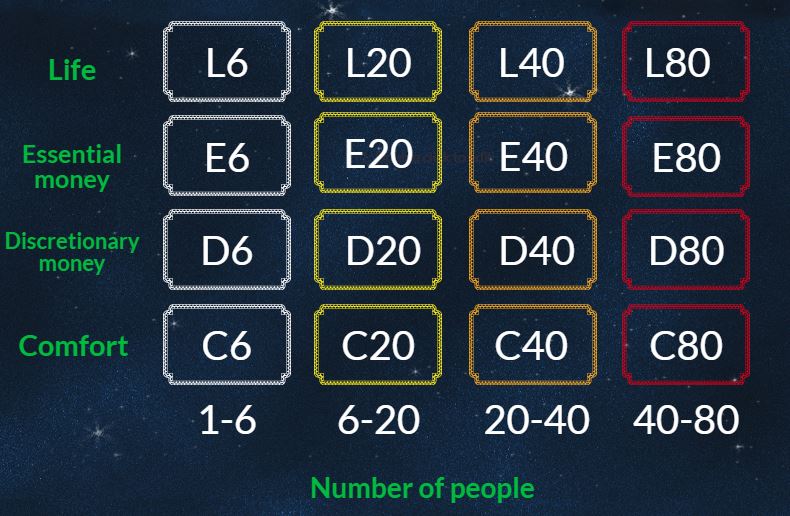
You may be wondering — what is the best Agile project management methodology: XP, SCRUM or FDD? People who’ve witnessed the most applications of each will say that none triumphs overall because different Agile methods suit different environments.

Any selection that you might make should ultimately depend on the demands and processes that factor into your business culture. While a certain method might work, it’s hard to fathom the changes that can occur in the world of information technology. However, the adaption of an Agile methodology can generate new levels of flexibility and productivity among development teams.

**V. Crystal Methodology**

The Crystal methodology is actually a family of smaller agile methodologies such as Crystal clear, Crystal yellow, Crystal red and etc. This set of agile methodologies is introduced by [Alistair Cockburn](http://alistair.cockburn.us/) who actually participated in writing the [Agile manifesto for software development](http://agilemanifesto.org/).

There are 3 main factors that will determine the characteristics of different projects: team size, system criticality, and project priorities. Projects are categorized depending on the system criticality as there are four levels of criticality: Comfort (C), Discretionary Money (D), Essential Money (E), and Life (L).

[](https://kanbanize.com/blog/wp-content/uploads/2015/11/Crystal-method-1.jpg)

The maximum number of people that need to be involved in a project depends on the size of the project. Bigger projects, more people. The number of team roles also depends on the project’s size. If the project is huge there are many different roles and vice versa. Additionally, these agile methodologies are focused on:

1. Interaction
2. People
3. Community
4. Skills
5. Communications
6. Talents

As most of the other agile methodologies, Crystal preaches early delivery of working software, frequency, less bureaucracy and high involvement of users. The Crystal family suggests that each project is unique and requires the application of different processes, practices, and policies. This is why it is perceived as one of the most lightweight agile methodologies.

**VI. Kanban Method (Agile or Lean)**

Kanban is definitely closer to Lean. However, it is probably more agile than most Agile methodologies.

[Kanban](https://kanbanize.com/kanban-resources/getting-started/what-is-kanban/) is a scheduling method developed by Toyota engineer Taiichi Ohno for Lean production. Designed as a system for scheduling, Kanban facilitates production and inventory control. Acclaimed for its ability to yield vast quantities of product, Kanban is one of the foremost methodologies through which work teams can accomplish [just-in-time (JIT) production](https://kanbanize.com/lean-management/pull/just-in-time-production/).

Kanban is a powerful method for facilitating entire systems of production and an effective means for generating improvements. A key benefit of Kanban is the way it limits the amount of work that can linger in an unfinished state, which helps prevent production chains from getting backlogged.

The roots of Kanban date back to the late 1940s, when Toyota was brainstorming ways to mimic the shelf-stocking methodologies of supermarkets in a factory setting. When a customer goes to a supply store, for instance, that customer will acquire the items that are needed.



A supply store only needs to be stocked with items that are sure to sell during a specific span of time. Since the items are sure to be there when a customer is liable to want them, no one needs to hoard for the future. However, in order to guide customers along through the shopping process, signs are placed everywhere to help people find what they need and make their purchases in an efficient, hassle-free manner.

Based on this analysis of the supermarket model, Toyota began to see production processes as metaphors for customers — just like customers go to stores in order to stock up on everyday essentials, a process relies on preceding processes for the ability to implement itself. As such, the process that comes before the current one is a store.

Kanban synchronizes quantity of stock with the viability of a given product. The supplier is informed by a signal that new product must be produced and shipped once a previous batch has been consumed. Each signal is followed along the replacement chain, which keeps people informed on both the supply side and consumer end. Essentially, Kanban bases its production frequency on consumption rates.

**Kanban: The Six Rules of Operation**

Ohno firmly believed that in order for the [Kanban method](https://kanbanize.com/kanban-resources/getting-started/what-is-kanban/) to be successful, companies had to follow its procedures thoroughly at all times, without exceptions.

On that note, he employed the six following rules at Toyota:

* 1. Latter process takes a quantity of supplies that have been specified by Kanban at the preceding process
  2. Early process makes products according to the amount and order that Kanban specifies
  3. Products are never to be manufactured or delivered with no Kanban
  4. A Kanban must be attached to all products
  5. Faulty items are never to be green-lit for the next process
  6. Sensitivity is raised by lower quantities of Kanban

**Kanban Cards**

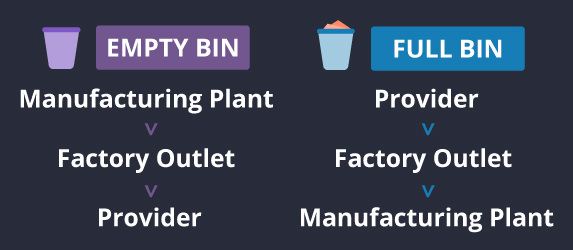
Kanban cards are one of the vital parts of the whole Kanban methodology, as they signify the necessity to direct products within a manufacturing plant. Kanban cards are memos that indicate the sell-out of supplies, parts or whole entire inventories.

Once it arrives, the Kanban card spurs a new production run of those supplies or parts. Ultimately, those on the consumer end control production with their wallets’ vote, and that vote is signified by the Kanban card — this puts the card at the heart of the consumer-driven paradigm.

**Three-Bin System**

The three-bin system is one of the more exemplary executions of the Kanban methodology. In this setup, a bin is placed at three different points: the manufacturing plant, the factory outlet, and the provider. Respectively, the three points represent the spots where demand, inventory control, and consumption take place. Each bin contains a detachable Kanban card that lists the specifications of the product in question.

Once the factory bin has been emptied through a manufacturing round, the bin is taken to the point of inventory control. From the factory outlet, a new bin is sent to the manufacturing plant while the empty bin is sent to the provider, who then sends a full bin of products to the outlet. Thus, it’s an ongoing cycle that runs as follows:



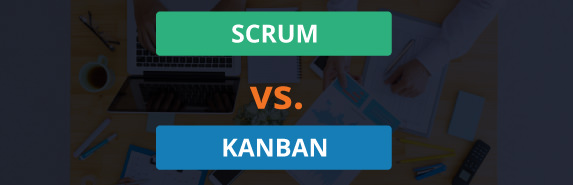
Each bin contains a Kanban card. In factories that employ the methodology, the color-coded box system is most often used.

**Electronic Kanban**

Numerous companies have put an electronic form of the Kanban methodology into practice. In doing so, these companies have eradicated the various types of errors that frequently occur with no such system in place — such as misplaced cards and manual entry mistakes.

The e-Kanban system, as it is known, can work with other management software systems to send signals along the manufacturing chain for any given company in real time. The information gathered via a [digital Kanban board](https://kanbanize.com/kanban-resources/kanban-software/digital-kanban-board/) is often helpful for the maximization of stock because the system makes it easier to monitor provider leads and refill cycles.

**VII. SCRUM vs. Kanban**



A lot of people see significant value in the concepts advanced by SCRUM, such as team organization and ongoing feedback. However, people in the know are now switching over to Kanban, a methodology that many say takes agility to a whole new realm and that truly harnesses the insights gained from Lean.

First, we need to point that SCRUM is an agile software development methodology. Or at least it is widely used by developers. In all fairness, SCRUM still has plenty of benefits. It offers a direct outline of what must be done amongst a team, it keeps things directed on an end goal and it helps teams decide the most efficient ways for getting things accomplished. SCRUM even offers a namesake master, who ensures that teams have the necessary tools to carry out a project and eliminate all possible roadblocks.

One of the most loved features of SCRUM is the feedback loops, which team members can check to ensure that things stay on track. Overall, the method offers a framework that helps users pinpoint where a software has gone wrong so they can gain new knowledge and adjust as needed.

An oft-raised complaint about SCRUM is the length of its time-boxed sprints, which are considered too long when employed with startups. The main criticism is that lengthy sprints lead to infrequent releases, which can cause work teams to drag their feet when responding to the needs of customers. This will often lead to counterproductive habits, such as teams working off of archaic knowledge.

***See also*** [Kanban vs Scrum Infographic](https://kanbanize.com/blog/kanban-vs-scrum-infographic/)

Then again, with undersized sprints, the larger features have to be broken up into smaller jobs, which are unlikely to be of use to a customer and might end up confusing the goals of the team. The set lengths of the SCRUM sprints were designed to offer consistency, and aren’t as useful in a world where technological innovations move at a faster rate than before. As innovations hit the market more frequently, people’s preferences are evolving with greater speed.

In order to remain on top of the market, developers must stay on top of this accelerated pace. Developers only stand to lose when the length of a sprint prevents tasks from ending in a timely manner.

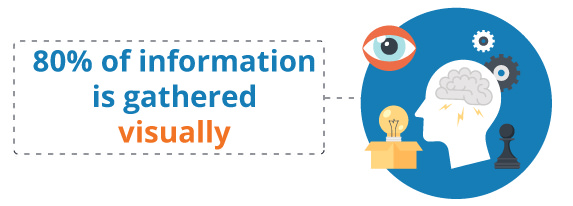
Kanban tackles the problems raised by SCRUM with a different scheduling protocol. Instead of operating with time-boxed sprints, Kanban restricts the number of things that a collective can focus on during any particular time span.

Once a feature has been finished, Kanban offers two possibilities:

* + - The feature is ready to be green-lighted for mass assembly (if those responsible opt to proceed).
    - The team is able to move to the project of second-most importance, which might be discovered that very same day.

**Kanban: The Visual Appeal**

On a standard Kanban board, there will be at least three columns with the following labels: To Do, Doing, and Done. Under each column is a limited set of colored notes that signify the tasks assigned to the given column. As studies have noted, [80 percent of information is gathered visually](http://venturebeat.com/2013/09/02/why-kanban-is-the-cure-to-your-companys-multitasking-plague/), which makes the Kanban board a powerful tool for noticing and remembering the things that must be done.



The columns can also be given alternate names if doing so better suits the team — the number of columns can even be extended beyond three, though it’s generally not advised to add too many because that could bog down the overall process. Although there are [Kanban board examples for different occasions](https://kanbanize.com/kanban-resources/kanban-software/kanban-board-examples/), the main objective is to splice the workload into several different, strictly defined stages. The key benefits of the method are as follows:

* + - Kanban helps teams stipulate varying task limits for different stages of a project.
    - Kanban offers an image-based timetable for the productivity of a team, which helps everyone involved collectively optimize and steer clear of any roadblocks that might appear.

The outcome of Kanban is twofold: You get more response from the marketplace, and you’re able to adjust to the demands of that input with greater agility.

While SCRUM does contain vital benefits — like ongoing feedback and the ability for teams to organize independently — these benefits are effectively usurped by the arrangement features of Kanban. Kanban marks the latest stage in Agile evolution — it’s a methodology that gathers the best practices from its predecessors for the challenges of tomorrow.

[](https://kanbanize.com/sign-up/)

**VIII. Bonus: Lean**

People often mistake Agile with Lean and vice versa. Both are very similar, but not the same thing.

Lean is short for [Lean Manufacturing](https://kanbanize.com/lean-management/what-is-lean-management/), which is a set of managerial principles that the Japanese developed in the middle of 20th century in order to ensure value and efficiency in production, while the Agile Manifesto was created in 2001. The original five principles of Lean management are as follows:

* Identify Value
* Value Stream Mapping
* Create Continuous Workflow
* Create a Pull System
* Continuous Improvement

The premise was later assimilated by Agile software developers Mary and Tom Poppendieck, to create what is today known as Lean Software Development. The concept was broken down into the following principles:

* Eliminate waste
* Build quality in
* Create knowledge
* Defer commitment
* Deliver fast
* Respect for people
* Optimize the whole

According to Tom Poppendieck, [the Agile software](https://www.infoq.com/interviews/poppendieck-interview) has always focused on delivering value, whatever that is. However, it hasn’t thought much at all about what it takes to deliver the value in terms of understanding capacity, in terms of developing capabilities, in terms of invoking the passion. All things that the Lean concept covers.

[](https://kanbanize.com/lean-software-development-with-kanban/)

The Lean thinking encourages businesses to dispense with all practices that hinder progress and instead maximize the strategies that work best for a given plan.

In order to eradicate waste, a company cuts out useless documents, unproductive tasks and time-consuming meetings that don’t forward the productivity of an initiative. Similarly, Lean entails focusing on things that need to be done in the here and now, and discourages preoccupations with anything that might be useful at a future point in time.

Another thing that Lean also stresses on is the manner by which teams collectively function, which is referred to in Lean parlance as “the system.” According to this, [respect for workers](https://kanbanize.com/blog/why-respect-for-people-is-the-secret-ingredient-of-successful-teams/) is a paramount, because each individual within a team of workers will best know his or her role within that team. Therefore, it’s most productive to trust each worker’s performance, while at the same time ensuring that the necessary tools are provided to make it all possible.

We hope that we brought clarity with this article, so now you can pick up the method that fits best your needs.

If you feel confident about Kanban, discover its real power with [Kanbanize](https://kanbanize.com/sign-up/).

**Related posts:**

1. [Stop Abusing Agile](https://kanbanize.com/blog/stop-abusing-agile/)
2. [3 Ways to Approach Lean, Agile, and Lean-Agile](https://kanbanize.com/blog/lean-agile-approach/)
3. [Myths about Kanban and Its Usage in the Software Development Process](https://kanbanize.com/blog/myths-about-kanban-for-software-development/)
4. [How to Start a Lean Agile Transformation? LAMP ep.5](https://kanbanize.com/blog/how-to-start-lean-transformation/)